ガウス過程回帰を用いたエンジンノックのモデル化とその制御

大 山 博 之*・山 北 昌 毅**
佐 多 宏 太***・松 永 彰 生***

Modeling and Control of Engine Knock Using Gaussian Process Regression

Hiroyuki OYAMA*, Masaki YAMAKITA**,
Kota SATA*** and Akio MATSUNAGA***

Knock limit control is necessary for high efficient operation of automotive engines. In this paper, we propose an identification and control method of the engine knock using a Gaussian process. Feedforward input is designed by the Gaussian process model approximating knock probability, and its model is updated online by a feedback of the knock intensity value. As modeling methods using the Gaussian process, we apply a heteroscedastic Gaussian process and a recursive Gaussian process. The effectiveness of the proposed method is demonstrated by a numerical simulation.

Key Words: Gaussian process, automotive engine, identification

1. はじめに

環境保全やエネルギー枯渇対策問題への関心が高まる中、自動車エンジンの開発においても、環境負荷低減などの社会的要求が高くなっている。そのため、高い燃料効率を実現し、排気ガス規制の要求を満たすエンジンの開発が課題となる。

一方で、自動車エンジンの制御は最適化が進んでおり、さらなる性能向上のためには衝撃波を伴う異常燃焼であるノック(ノッキング)や失火といった異常運転を起こす領域近傍での制御が要求される。たとえば、点火時期はエンジンの性能を決定する重要な制御入力の一つであるが、ガソリンエンジンにおいて出力トルクに対して最適となる点火時期 (Minimum advance for the Best Torque, MBT) はノックが頻繁に発生する運転領域となる場合がある¹⁾。エンジン運転の効率化のためにはノックの発生率が十分に小さくなる運転領域で、点火時期を MBT にできるだけ近づける制御が望まれ、その制御はノックリミット制御と呼ばれる。ノックは同じ運転条件で

も確率的に発生することが知られており^{2),3)}、ノックリミット制御では統計的制御手法によりノック発生確率を目標値とするような制御が行なわれる^{4)~8)}。

本稿では、筒内圧波形から算出したノック強度値と点火時期の関係を統計的モデルであるガウス過程回帰⁹⁾を用いてモデル化し、得られた確率モデルからフィードフォワード入力を求める高精度なノックリミット制御手法を提案する。まず、異分散ガウス過程を用いたモデルの学習について述べた後に、オンラインでのモデル更新を目的として局所的に逐次ガウス過程¹⁰⁾により近似する手法を提案する。その有効性は実験データから作成した数理モデルを用いた数値シミュレーションにて確認される。

本稿はつぎのように構成される。2章では作成した数理プラントモデルと従来の制御手法について説明する。3章ではガウス過程を用いた確率モデルの学習と、獲得したモデルを用いた制御手法を提案する。4章では提案手法の有効性を数値シミュレーションを用いて示す。最後に5章で結論と今後の課題についてまとめる。

2. 従来手法

本章では数値シミュレーションに用いるエンジンノックについて説明した後に、従来のノックリミット制御を紹介する。ガソリンエンジンシステムは多入力システムであるが、本稿では特にエンジンノックに大きな影響を与える点火時期のみをノックリミット制御の入力として扱う。簡単化のためにスロットル開度やバルブリフト量といった点火時期以外の制御入力は一定とし、点火時期を入力、ノック発生確率を出力とした1入力1出力システムの制御問題を考える。点火時期は

* 東京工業大学機械制御システム専攻
東京都目黒区大岡山 2-12-1

** 東京工業大学システム制御系 東京都目黒区大岡山 2-12-1

*** トヨタ自動車(株)東富士研究所 裾野市御宿 1200

* Dept. of Mechanical and Control Engineering, Tokyo Institute of Technology, 2-12-1 Ookayama, Meguro-ku, Tokyo

** Dept. of Systems and Control Engineering, Tokyo Institute of Technology, 2-12-1 Ookayama, Meguro-ku, Tokyo

*** Higashi-fuji Technical Center, Toyota Motor Corporation, 1200 Mishuku, Susono

(Received May 24, 2018)

(Revised March 19, 2019)

Fig. 1 Characteristic of torque as a function of spark advance

Fig. 2 Structure of a conventional knock limit control system

圧縮上死点を基準としたクランク角度で表わされるが、本稿では上死点からどれだけ早く点火を行なったかという点火進角 (Spark Advance, SA) を用いて表わす。また、点火時期を早めることを進角、点火時期を遅らせることを遅角と表現する。Fig. 1 は制御入力を点火時期とした場合のエンジントルク曲線とノック境界との関係を示した図である。ノックが発生しやすいノック境界近傍では通常 MBT に近づくほどノックの発生確率は高くなる。制御入力を MBT にできるだけ近づけるには、許容されるノック確率を達成するよう制御すればよく、ノックリミット制御では目標値をノック発生確率とした統計的制御器による制御を行なう。

典型的な統計的制御器^{4), 5), 7), 8)}へのフィードバック信号は、ノック発生の検知結果であり、ノックセンサが計測したノック信号からノックの有無が判別される。また、最近ではノック信号から算出したノック強度をフィードバック信号とする制御器が提案されており⁶⁾、本稿でもノック強度をフィードバック信号として扱う。Fig. 2 にノック強度をフィードバックとするシステムの構成例を示す。

2.1 数値シミュレーションのプラントモデル

ここでは、数値シミュレーションのプラントモデルとして用いる、実験データから作成したエンジンノックモデル Fig. 3～Fig. 5 (以下、ノックモデルと呼ぶ) について説明する。ノックモデルは入力を点火時期、出力をエンジンのノック強度とした静的モデルであり、出力は確率密度関数として表わされる。ノックの検知は、筒内圧波形からノック強度値を算出し、ノック強度がある閾値 KI を超えた場合にノックが発生したと判断される。ノックは同じ運転条件下でも確率的に発生することが知られており、特にノック強度は対数正規分布でよく近似される^{2), 3), 8)}。本稿の実験データは文献 8) で扱っている 6 気筒エンジンベンチ環境を用いて計測したものであり、文献 8) の Fig. 5, Fig. 6 と同様にそのノック強度は対数正規分布でよく近似される。

Fig. 3 Mean value data of engine operation

Fig. 4 Variance value data of engine operation

Fig. 5 Numerical model of knock probability (3D view)

ノックモデルでは点火時期以外の運転条件を一定にした上で点火時期 1 deg ごとのノック強度の対数値データ 1500 点を正規分布で近似し、その後、平均値と分散値を 3 次スプライン曲線で近似したものを点火時期のサンプリング間データとして内挿している。以下、ノック強度の対数値を対数ノック強度と呼ぶこととする。Fig. 3 は点火時期に対する対数ノック強度の平均値を、Fig. 4 は対数ノック強度の分散値を示しており、図のサークルは実験データ、実線は 3 次スプライン曲線による近似結果である。また、Fig. 3, Fig. 4 の結果を点火時期に対する対数ノック強度の確率密度関数として視覚化したものが Fig. 5 であり、3 次元プロットで表わしている。

この後の数値シミュレーションにおいては、点火時期入力 が 0.1 deg ごとの値をとれるとし、小数第一位で丸めた入力値を実際の入力として使用する。また、ノック検知の閾値は $KI = \exp(4.5)$ とする。

2.2 アドバンス・リタード制御

ノックリミット制御のうち最も一般的な統計的制御器は、点火時期の進角 (アドバンス) と遅角 (リタード) を繰り返しながらノック発生率を調整するアドバンス・リタード制御⁴⁾である。その制御則はつぎのように表わすことができる。

$$esa_{t+1} = \begin{cases} esa_t - K_{Ret}, & \text{if knocking} \\ esa_t + K_{Adv}, & \text{otherwise} \end{cases} \quad (1)$$

ここで、 t はサイクル数、 esa_t はそのサイクルの点火時期指令値、 K_{Adv} と K_{Ret} はそれぞれ進角と遅角時のゲインであり、これら 2 つのゲインによりノック発生確率が定まる。たとえば、目

Fig. 6 Closed-loop spark angle response using an advance/retard control ($p_{ref} = 0.02$, $K_{Ret} = 1.2$, $K_{Adv} = K_{Ret}/49$)

Fig. 7 Histogram of spark angle response using an advance/retard control (10000 cycle)

標ノック発生確率 p_{ref} が与えられたとき、 $p_{ref} = \frac{K_{Adv}}{K_{Ret} + K_{Adv}}$ を満たすようゲインを設計することで、所望のノック発生確率が達成される⁵⁾。

ノックモデルに対して、この制御を適用した例を **Fig. 6**, **Fig. 7** に示す。Fig. 6 は各サイクルにおける制御入力である点火時期のようすを示しており、Fig. 7 は 10000 サイクルの制御を行なった場合の点火時期のヒストグラムを示している。この例では、 $p_{ref} = 0.02$ としているため、入力の境界は 11.8 deg として与えられ、理想的にはこの境界上の入力値をとることが望ましい。一方で、境界を越えた進角はノックによるエンジンへの損傷をもたらす可能性があるため望ましくないことに注意する。境界を越えた進角が発生した場合には、ノック発生後の素早い遅角が安全上重要である。アドバンス・リタード制御はノック発生確率は目標値に非常に良く追従するものの、遅角時のゲイン K_{Ret} を大きくした場合には入力値の分散が大きく、平均値も境界から大きく離れてしまう傾向がある。

2.3 尤度ベース制御

前述のアドバンス・リタード制御に対して、ノック強度の累積和^{5), 7)}や尤度^{5), 6)}、ノック強度の指数移動平均⁸⁾を用いた制御が提案されている。特に尤度ベースの制御ではつぎで示される尤度比を指標に制御が行なわれる。

$$L_n(KEC) = \frac{p_{ref}^{KEC} (1 - p_{ref})^{n-KEC}}{p_{max}^{KEC} (1 - p_{max})^{n-KEC}} \quad (2)$$

ここで、 n はカウントを開始してからサイクル数、 KEC はカウントを開始してからノック発生数であり、 $p_{max} = KEC/n$ である。尤度比をもとにした制御手法はいくつか提案されているが、その中でも文献 5) の Algorithm 3 を適用した結果を **Fig. 8**, **Fig. 9** に示す。Fig. 8 は各サイクルにおける点火

Fig. 8 Closed-loop spark angle response using a likelihood-based control ($p_{ref} = 0.02$, $K_{Ret} = 0.4$, $K_{Adv} = 0.4$, threshold = 0.4)

Fig. 9 Histogram of spark angle response using a likelihood-based control (10000 cycle)

時期のようすを示しており、Fig. 9 は点火時期のヒストグラムを示している。図からわかるように、尤度ベースの手法はアドバンス・リタード制御と比較し、応答速度を維持した上で、入力値の分散を小さくできる。一方で、ノック発生時の遅角が遅れる傾向があることと、ノックの発生確率が目標値に対して収束するようなゲイン設計法が確立しておらずゲイン調整が課題となる。

3. ガウス過程を用いたノック制御

2 章で紹介した従来手法はフィードバックによって目標となるノック確率に収束させる制御を行なう。しかしながら、ノックの発生確率およびノック境界は燃料の種類や湿度、エンジン水温などにより頻繁に変化する。その変化に追従するために従来制御では制御ゲインを大きくする必要があり、結果として入力値の分散が大きくなるという課題があった。この章では、ガウス過程 (Gaussian Process, GP) モデルを用いたフィードフォワード入力を利用する、高精度なノックリミット制御手法を提案する。一般に確率モデルの構築には膨大な数のデータが必要となるが、ノック強度の対数値の確率分布がガウス過程で良く近似できるという性質より、GP モデルを用いることで少量の学習データからモデル構築が可能となる。ノック強度の確率モデルを得ることができれば、ノックの発生確率は容易に計算できるため、目標ノック確率となる最適点火時期の推定値を求めることができ、フィードフォワード入力として与えられる。**Fig. 10** は提案手法のシステム構成を表わしたものである。ここではエンジンノック強度を筒内圧センサを用いて算出、フィードバックすることで逐次的に GP モデルを更新している。また、フィードバック制御則として 2 章で紹介した従来手法を組み合わせることで応

Fig. 10 Structure of a proposed knock limit control system

Algorithm 1 GP-based control with advance/retard control

- 1: **Initialization:** Initialize parameters.
- 2: **Preparation:** Acquire initial training input-output data set $\mathcal{D}_0$.
- 3: **Closed-loop control**
- 4: For $t = 0, \dots, t_{end}$
- 5: Update GP model with training data $\mathcal{D}_t$ and calculate a knock probability $p_t(esa)$.
- 6: Determine a feed forward input esa_t^* as,
 $esa_t^* = \max_{esa} p_t(esa) < p_{ref}$.
- 7: Apply the input $esa_t^* + \Delta_t$ to the system and obtain a new training data set $\mathcal{D}_{t+1}$.
- 8: Update a feed back input as,

$$\Delta_{t+1} = \begin{cases} \Delta_t - K_{Ret}, & \text{if knocking,} \\ \Delta_t + K_{Adv}, & \text{otherwise.} \end{cases}$$
- 9: End

答速度の向上とデータの偏りによる学習モデルの精度低下防止を図る。アドバンス・リタード制御を用いた提案手法の概要を Algorithm 1 に示した後、適用する GP モデルについて述べる。

3.1 ガウス過程

GP はカーネル法をベイズ確率理論に基づいて解析したものであるが⁹⁾、ここでは回帰問題を考える。GP 回帰の目的は、 d 次元の入力ベクトル $\mathbf{x}_*$ が与えられたときにそれに相当する未知の出力 y_* を予測することである。 N 組の学習データを考えたとき、入力学習データを $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]^T$ 、出力の学習データを $\mathbf{y} = [y_1, y_2, \dots, y_N]^T$ 、学習データを $\mathcal{D} = (\mathbf{X}, \mathbf{y})$ とする。あるカーネル関数 $k(\cdot, \cdot)$ を用いて GP の事前分布を $f(\mathbf{x}) \sim \mathcal{GP}(\mathbf{0}, k(\mathbf{x}, \mathbf{x}'))$ 、観測ノイズが平均値 0、分散値 σ^2 のガウス分布に従うとしたとき、 $y|\mathbf{x} \sim \mathcal{N}(f(\mathbf{x}), \sigma^2)$ と書け、予測分布は以下のように表現される^(注 1)。

$$y_*|\mathbf{x}_*, \theta, \mathcal{D} \sim \mathcal{N}(\mu_{f*}, \sigma_{f*}^2 + \sigma^2 \mathbf{I}) \quad (3)$$

$$\mu_{f*} = k(\mathbf{x}_*, \mathbf{X})(\mathbf{K} + \sigma^2 \mathbf{I})^{-1} \mathbf{y} \quad (4)$$

$$\sigma_{f*}^2 = k(\mathbf{x}_*, \mathbf{x}_*) - k(\mathbf{x}_*, \mathbf{X})(\mathbf{K} + \sigma^2 \mathbf{I})^{-1} \times k(\mathbf{X}, \mathbf{x}_*) \quad (5)$$

ただし、 $\mathbf{K} = k(\mathbf{X}, \mathbf{X})$ かつ、行列 $\mathbf{X}$ が与えられたとき、カーネル行列 $k(\cdot, \cdot)$ は以下で定義される。

$$k(\mathbf{X}, \mathbf{X}) = [k_{ij}], \quad k_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$$

(注 1) $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ は平均ベクトルが $\boldsymbol{\mu}$ 、共分散行列が Σ のガウス分布を表わす。

$$k(\mathbf{x}^*, \mathbf{X}) = [k(\mathbf{x}^*, \mathbf{x}_1), \dots, k(\mathbf{x}^*, \mathbf{x}_N)] \\ = k(\mathbf{X}, \mathbf{x}^*)^T$$

GP は主にカーネル関数 $k(\mathbf{x}, \mathbf{x}')$ によりその性質が定まる。本稿では、以下のような関連度自動決定 (Automatic Relevance Determination, ARD) カーネルを用いる。

$$k(\mathbf{x}, \mathbf{x}') = \text{cov}\{f(\mathbf{x}), f(\mathbf{x}')\} \quad (6)$$

$$= \lambda^2 \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{x}')^T \Lambda^{-1}(\mathbf{x} - \mathbf{x}')\right) \quad (7)$$

ここで、 $\Lambda = \text{diag}(l_1^2, l_2^2, \dots, l_d^2)$ は $\mathbf{x}$ の各要素間の関係の特徴づけるパラメータであり、 λ^2 は潜在関数 f の分散を表わすパラメータである。すべてのパラメータ $\theta = [l_1^2, l_2^2, \dots, l_d^2, \lambda^2, \sigma^2]$ はハイパーパラメータと呼ばれ、以下の周辺尤度最大化より最適なパラメータを見つけることができる。

$$\theta_{best} = \underset{\theta}{\text{argmax}} \log(p(\mathbf{y}|\mathbf{X}, \theta)) \quad (8)$$

$$\log(p(\mathbf{y}|\mathbf{X}, \theta)) = -\frac{1}{2} \mathbf{y}^T (\mathbf{K} + \sigma^2 \mathbf{I})^{-1} \mathbf{y} \\ - \frac{1}{2} \log |\mathbf{K} + \sigma^2 \mathbf{I}| - \frac{N}{2} \log 2\pi \quad (9)$$

3.2 異分散ガウス過程

通常、GP において観測ノイズの分散値 σ^2 は入力値に依存しないスカラー値であるため、入力値に依存する分散をもつノックモデルは表現できない。そこで、入力値に依存する分散を表現できるよう、つぎのようなノイズモデルを追加した異分散ガウス過程 (Heteroscedastic Gaussian Process, HGP) を考える^{11)~14)}。

$$y|\mathbf{x} \sim \mathcal{N}(f(\mathbf{x}), \tilde{\sigma}^2(\mathbf{x})) \quad (10)$$

$$v = \log(\tilde{\sigma}^2(\mathbf{x})) \sim \mathcal{GP}(\tilde{m}(\mathbf{x}), \tilde{k}(\mathbf{x}, \mathbf{x}')) \quad (11)$$

ここで、観測ノイズは平均値 0、分散値 $\tilde{\sigma}^2(\mathbf{x})$ のガウス分布に従うとしており、 $\log(\tilde{\sigma}^2(\mathbf{x}))$ の分布は平均値関数を $\tilde{m}(\mathbf{x})$ 、カーネル関数を $\tilde{k}(\mathbf{x}, \mathbf{x}')$ としたガウス過程で表わす。ノイズモデルのハイパーパラメータを $\tilde{\theta}$ とすると、HGP のハイパーパラメータ ϕ は $\phi = [\theta, \tilde{\theta}]$ となり、たとえば期待値伝播法を応用することで学習により求まる¹²⁾。詳細は省くが、この学習では、 v の事後分布 $p(v|\mathcal{D})$ をガウス分布 $q(v|\mathcal{D})$ に近似し、周辺尤度最大化によりハイパーパラメータ ϕ を求める。

入力 $\mathbf{x}_*$ を与えたときの出力の予測値 y_* はガウス分布で近似した $q(v_*|\mathbf{x}_*, \mathcal{D}) = \mathcal{N}(\mu_{v*}, \sigma_{v*}^2)$ を用いて以下で計算できる。

$$y_*|\mathbf{x}_*, \phi, \mathcal{D} \sim \mathcal{N}(\mu_*, \sigma_*^2) \quad (12)$$

$$\mu_* = \mu_{f*} \quad (13)$$

$$\sigma_*^2 = \sigma_{f*}^2 + \exp(\mu_{v*} + 0.5\sigma_{v*}^2) \quad (14)$$

HGP によりノックモデルを学習する場合、ノイズモデルのカーネル関数 $\tilde{k}(\mathbf{x}, \mathbf{x}')$ を $\tilde{k}(\mathbf{x}, \mathbf{x}') = \tilde{\lambda}^2 \exp(-\frac{1}{2}\|\mathbf{x} - \mathbf{x}'\|/\tilde{l}_1^2)$ 、観測ノイズが平均値 0、分散値 σ_{noise}^2 のガウス分布に従うとすると、HGP のハイパーパラメータは $\phi =$

Table 1 Comparison result between GP and HGP models. The loss between the predicted value and the true value is evaluated by RMSE. Each result uses 100 training data sets and mean is shown $\pm$ one standard deviation

Method	N	Mean loss [deg]	Variance loss [deg ²]
GP	200	0.0903 $\pm$ 0.0476	0.1240 $\pm$ 0.0170
	300	0.0811 $\pm$ 0.0347	0.1252 $\pm$ 0.0136
	400	0.0679 $\pm$ 0.0260	0.1242 $\pm$ 0.0098
	500	0.0698 $\pm$ 0.0282	0.1236 $\pm$ 0.0090
HGP	200	0.1204 $\pm$ 0.0419	0.0959 $\pm$ 0.0239
	300	0.0952 $\pm$ 0.0367	0.0741 $\pm$ 0.0204
	400	0.0873 $\pm$ 0.0284	0.0661 $\pm$ 0.0193
	500	0.0824 $\pm$ 0.0330	0.0636 $\pm$ 0.0179

Fig. 11 Example of heteroscedastic Gaussian process regression (300 data points)

$[\hat{l}_1^2, \hat{l}_2^2, \hat{\lambda}^2, \hat{\lambda}^2, \hat{\sigma}^2, \hat{\sigma}_{noise}^2]$ と書ける. **Table 1** には, 学習された GP モデルと HGP モデルの予測平均と予測分散の平均平方二乗誤差を学習データ点数 N ごとに比較している. GP モデルでは学習データ点数の増加に対し予測分散の精度が向上しない一方で, HGP モデルでは予測分散の精度が学習データ点数の増加に従い向上しており, GP モデルと比較して良い結果を示していることがわかる. これは, HGP モデルが GP モデルと比較して柔軟なノイズモデルの表現ができるためであると考えられる. また, **Fig. 11** に学習データ 300 点を用いた HGP モデルの例を示す. ノックモデルに対し, HGP モデルによる予測結果をメッシュで 3 次元プロットしている.

3.3 逐次ガウス過程を用いた局所モデル

HGP はノックモデルの学習に適しているものの, 学習に必要な計算コストは小さくないためオンライン更新には向かない. そのため, ノック境界の変化に対応が難しく, 制御への応用には不向きである. 一方で, ノック境界近傍のみを考える局所モデルでは, 観測ノイズがスカラー値である GP モデルで十分な近似が可能であると考えられる. つまり, 従来手法や異分散ガウス過程モデルによりノック境界の大まかな推定ができれば, 詳細なノック境界は計算コストが小さくオンライン更新に適した GP により求めることができ, ノック境界の変化にも適応可能となる.

ここでは高速なガウス過程のオンライン学習手法として逐次ガウス過程 (Recursive Gaussian Process, RGP) の適用

を考える. RGP はガウス過程の近似手法の 1 つであり¹⁰⁾, オンラインに更新を行ない, 関数を推定する. まず, GP と同様に, 学習データ $\mathcal{D} = (\mathbf{X}, \mathbf{y})$ を用いて $f = f(\mathbf{X})$ を定義する. f が初期分布 $p(f) = \mathcal{N}(f|\mu_0^f, C_0^f)$ をもつ GP であると仮定する. ここで $\mu_0^f = 0$, $C_0^f = k(\mathbf{X}, \mathbf{X})$ であり, これは GP における事前分布と等しい. GP では, 一度定義された事前分布は変わらないが, RGP における事前分布は, 新たな入力 $\mathbf{x}_t$ とそれに対応する観測値 y_t が与えられるとオンラインで更新される. その更新式はカルマンフィルターの更新則と同様に, 以下の 2 つの計算により行なわれる.

Prediction: $t-1$ サイクルの事後分布を用いて, t サイクルでの予測分布 $p(y_t|\mathbf{y}_{1:t-1}) = \mathcal{N}(y_t|\mu_t^p, C_t^p + \sigma^2\mathbf{I})$ を計算する.

$$\mu_t^p = \mathbf{J}_t \mu_{t-1}^f \quad (15)$$

$$\mathbf{B}_t = k(\mathbf{x}_t, \mathbf{x}_t) - \mathbf{J}_t k(\mathbf{X}, \mathbf{x}_t) \quad (16)$$

$$\mathbf{J}_t = k(\mathbf{x}_t, \mathbf{X}) k(\mathbf{X}, \mathbf{X})^{-1} \quad (17)$$

$$\mathbf{C}_t^p = \mathbf{B}_t + \mathbf{J}_t \mathbf{C}_{t-1}^f \mathbf{J}_t^T \quad (18)$$

Update: 新たな観測値 y_t を用いて, f の事後分布は以下で求められる.

$$\mu_t^f = \mu_{t-1}^f + \mathbf{G}_t (y_t - \mu_t^p) \quad (19)$$

$$\mathbf{C}_t^f = \mathbf{C}_{t-1}^f - \mathbf{G}_t \mathbf{J}_t \mathbf{C}_{t-1}^f \quad (20)$$

$$\mathbf{G}_t = \mathbf{C}_{t-1}^f \mathbf{J}_t^T (\mathbf{C}_t^p + \sigma^2\mathbf{I})^{-1} \quad (21)$$

GP の性質はハイパーパラメータにより定まるが, 適用しようとしている RGP はハイパーパラメータの更新は行っていないことに注意する. また, 学習に必要な計算コストは基底となる学習データ $\mathcal{D}$ に依存するため, 新たに学習データが追加されても計算コストは増加しない. ハイパーパラメータも同時に更新する逐次アルゴリズム¹⁰⁾も提案されているものの, 計算コストが大きくエンジンシステムのような更新周期が短いオンライン学習には適さない. つまり, この逐次学習の利点は, 計算コストが大きいハイパーパラメータの更新を避けながらも, 初期学習データを補正することにより, 予測精度を向上できる点にある.

4. 数 値 例

本章では, ノックモデルを用いた数値シミュレーションにより提案手法の有効性を確認する.

4.1 異分散ガウス過程を用いた制御

学習した HGP モデルを用いたフィードフォワード入力のみを用いた制御結果を **Fig. 12**, **Fig. 13** に示す. **Fig. 12** は初期学習データとしてノックモデルより生成された 150 点の学習データを用いた場合の, 各サイクルでの点火時期 esa のようすを, **Fig. 13** は 200 サイクルまでの点火時期の分布をヒストグラムとして示している. HGP モデルは獲得したノック強度データをもとに適応的に変化するため, 点火時期はノック境界に収束していくことがわかる. これが, 常に入力値が

Table 2 Steady-state knock controller performance in 200 cycles. Each result uses 100 trials and mean is shown $\pm$ one standard deviation

Method	Initial SA (wrt boundary) [deg]	Mean value of SA (wrt boundary) [deg]	Variance value of SA [deg ²]	Max. value of SA (wrt boundary) [deg]	Knock rate
Advance/retard control	-1.0	-0.3067 ± 0.3529	0.4972 ± 0.3567	1.2011 ± 0.5794	0.0185 ± 0.0016
($p_{ref} = 0.02$, $K_{Ret} = 1.2$, $K_{Adv} = K_{Ret}/49$)	0.0	-0.2319 ± 0.2855	0.5418 ± 0.3328	1.3543 ± 0.5104	0.0206 ± 0.0017
	1.0	-0.1156 ± 0.3495	0.6608 ± 0.4806	1.6588 ± 0.4382	0.0223 ± 0.0018
Likelihood-based control	-1.0	-0.1882 ± 0.1955	0.2706 ± 0.1228	0.6938 ± 0.3228	0.0183 ± 0.0052
($p_{ref} = 0.02$, $K_{Ret} = 0.4$, $K_{Adv} = 0.4$, threshold = 0.4)	0.0	0.3234 ± 0.3278	0.1481 ± 0.0870	1.0675 ± 0.4365	0.0290 ± 0.0057
	1.0	0.6987 ± 0.4093	0.1517 ± 0.0965	1.4536 ± 0.3884	0.0365 ± 0.0057
HGP-base control w/ advance/retard control	-1.0	0.0475 ± 0.2020	0.0588 ± 0.0524	0.3860 ± 0.1370	0.0252 ± 0.0093
($p_{ref} = 0.02$, 150 initial data)	0.0	0.0210 ± 0.2495	0.0778 ± 0.0654	0.4520 ± 0.1764	0.0244 ± 0.0088
	1.0	0.0638 ± 0.2070	0.0683 ± 0.0460	1.0 ± 0.0000	0.0234 ± 0.0089
RGP-base control w/ advance/retard control	-1.0	-0.0441 ± 0.1320	0.0272 ± 0.0316	0.1720 ± 0.0573	0.0209 ± 0.0066
($p_{ref} = 0.02$, 100 basis data)	0.0	0.0003 ± 0.1515	0.0232 ± 0.0277	0.2980 ± 0.1790	0.0225 ± 0.0053
	1.0	-0.0593 ± 0.1609	0.0182 ± 0.0114	1.1880 ± 0.1288	0.0202 ± 0.0062

Fig. 12 Closed-loop spark angle response using heteroscedastic Gaussian process model (the number of initial data = 150) **Fig. 14** Closed-loop spark angle response using local recursive Gaussian process model (the number of basis data = 100) **Fig. 13** Histogram of spark angle response (200 cycle)

変動する従来手法との大きな違いである。

また、従来手法との比較結果を **Table 2** に示す。ここでは点火時期をノック境界を基準としたときの差分として表わし、初期値を -1.0 deg, 0.0 deg, 1.0 deg と設定、各初期値に対して 200 サイクルまでの数値シミュレーションを 100 回試行している。初期値が -1.0 deg のシミュレーションは、ノック境界への進角の早さを、初期値が 1.0 deg のシミュレーションは、ノック境界を越えた場合の遅角の速さを、それぞれ検証することを目的としている。制御器の性能は点火時期の平均値と分散値、最大値およびノック発生確率によって評価している。点火時期の平均値はノック境界に近いほど、分散値および最大値は小さいほど望ましく、ノック発生頻度は目標値 (ここでは $p_{ref} = 0.02$) に近いほど望ましいといえる。HGP ベースの制御結果を従来手法の結果と比較すると、点火時期の平均値と分散値、最大値に関しては、従来手法と比較して性能の向上が見られる。これはノックが発生せずともノック強度からモデルおよび制御入力 (進角・遅角) が行なえ

ることと、ノック境界付近の正確なモデルが入手できれば点火時期がノック境界に収束していくことに起因していると考えられる。一方でノック発生確率はアドバンス・リタード制御が良い性能を示していることがわかる。これは、HGP ベースの制御においては境界近傍までは素早く到達するものの、境界近傍の正確なモデルを学習し、点火時期がノック境界に収束するまでにはある程度のサイクル数が必要になるためだと考えられる。以上のように提案する HGP ベースの制御は、従来手法と比較し良い性能を示すといえるものの、HGP モデルの計算コストは GP の学習と比較しても大きく、オンライン学習には向かないといった欠点がある。

4.2 局所逐次ガウス過程を用いた制御

RGP を用いた制御結果の例を **Fig. 14** に示す。Algorithm 1 で示したとおり、統計的制御器によるフィードバック制御として (1) 式に従った制御を追加している。ただし、 $K_{Ret} = 0.1$ とした。Fig. 14 では RGP のみを用いた制御と、RGP と統計的制御を組み合わせた制御に対して、ノックモデルを途中で変化した場合の追従性能を示している。両者とも境界の変化に対応できているが、この例では統計的制御を加えたものが良い応答性能を示していることがわかる。このとき、1 サイクル当たりの平均更新計算時間は MATLAB R2017a (Core i7-4770S, RAM: 16.0 GB) において 2.8 ms となった。

ほかの制御手法との比較結果を Table 2 に示す。ここで、RGP をベースにした手法では、ノック境界が初期値となるよ

うな分布の異なるノックモデル (たとえば, 初期値が -1.0 deg の場合にはノック境界が 10.8 deg となるような別のノックモデル) から生成したデータを保持した状態で制御をスタートしている. したがって, 初期値が -1.0 deg または 1.0 deg と設定されたシミュレーション検証結果は, ノックモデルの変化に対する提案手法のオンライン学習性能を示しているといえる. その上で従来手法と比較しても, はばすべての評価指標においてより良い性能を示していること, 特に点火時期の平均値と分散値においては, HGP ベースの制御と比較しても良い性能を示していることがわかる. 以上のことから, RGP ベースによる制御は境界近傍を特定する必要はあるものの, 高い性能とオンライン学習性能を両立しているといえる.

5. おわりに

本稿では, GP モデルを応用したノックリミット制御を提案した. 提案手法では少量のデータから GP モデルにより, ノックの確率モデルを学習する. 学習したモデルから目標ノック確率となる最適点火時期の推定値を計算し, フィードフォワード入力として与え, フィードバック制御則として従来の統計的制御器を組み合わせることで応答速度の向上とデータの偏りによる学習モデルの精度低下防止を図った. その後, 提案手法の有効性を実験データから作成した数理モデルをプラントモデルとした数値シミュレーションにて確認した. 学習する GP モデルとしては, HGP と RGP を用いた結果を示し, 特に局所的に RGP を適用することで, ノック境界の変化にも対応可能なオンライン学習手法を提案した. 今後の課題としては, 実機での運転条件変化を想定した定量的な評価, 基底学習データの合理的な選択手法の提案が挙げられる.

参 考 文 献

- 1) S. Watanabe and A. Ohata: Benchmark problem for near boundary operation control for automotive engine, Proc. 6th Conf. Simul. Test. Automot. Electron., 101/110 (2014)
- 2) J.M. Spelina, J.C. Peyton Jones and J. Frey: Characterization of knock intensity distributions: Part 1: Statistical independence and scalar measures, Proc. Inst. Mech. Eng. D: J. Automob. Eng., **228-2**, 117/128 (2013)
- 3) J.M. Spelina, J.C. Peyton Jones and J. Frey: Characterization of knock intensity distributions: Part 2: Parametric models, Proc. Inst. Mech. Eng. D: J. Automob. Eng., **227-12**, 1650/1660 (2013)
- 4) U. Kiencke and L. Nielsen: *Automotive Control Systems for Engine, Driveline, and Vehicle*, Springer (2005)
- 5) J.C. Peyton Jones, J.M. Spelina and J. Frey: Likelihood-based control of engine knock, IEEE Trans. Contr. Syst. Technol., **21-6**, 2169/2180 (2013)
- 6) A. Thomasson, L. Eriksson, T. Lindell, J.C. Peyton Jones, J. Spelina and J. Frey: Tuning and experimental evaluation of a likelihood-based engine knock controller, Proc. 52nd IEEE Conf. Dec. Contr., 6849/6854 (2013)
- 7) J. Frey, J.C. Peyton Jones and S. Shayestehmanesh: Stochastic simulation of a CumSum knock controller, Proc. 8th IFAC Int. Symp. Adv. Automot. Contr., 210/216 (2016)
- 8) X. Shen and T. Shen: Knock limit controller based on

exponential moving average of knock intensity, Proc. 8th IFAC Int. Symp. Adv. Automot. Contr., 691/695 (2016)

- 9) C.E. Rasmussen and C.K.I. Williams: *Gaussian Processes for Machine Learning*, The MIT Press (2006)
- 10) M.F. Huber: Recursive Gaussian process: On-line regression and learning, Pattern Recognition Letters, **45**, 85/91 (2014)
- 11) P.W. Goldberg, C.K.I. Williams and C.M. Bishop: Regression with input-dependent noise: A Gaussian process treatment, Adv. Neural Inf. Process. Syst., **10**, 493/499 (1997)
- 12) V. Tolvanen, P. Jylanki and A. Vehtari: Expectation propagation for nonstationary heteroscedastic Gaussian process regression, IEEE Int. Worksh. Mach. Learn. Signal Process., 1/6 (2014)
- 13) Q.V. Le, A.J. Smola and S. Canu: Heteroscedastic Gaussian process regression, Proc. Int. Conf. Mach. Learn., 489/496 (2005)
- 14) K. Kersting, C. Plagemann, P. Pfaff and W. Burgard: Most likely heteroscedastic Gaussian process regression, Proc. Int. Conf. Mach. Learn., 393/400 (2007)

[著 者 紹 介]

大 山 博 之 (正会員)

2015 年東京工業大学大学院理工学研究科機械制御システム専攻修士課程修了. 同年, 東京工業大学大学院理工学研究科機械制御システム専攻博士課程入学, 現在に至る. 制御工学, 自動車エンジン制御, モデル予測制御, モデル同定の研究に従事.

山 北 昌 毅 (正会員)

1989 年東京工業大学大学院理工学研究科制御工学専攻博士課程修了. 同年同大学工学部助手, 93 年豊橋技術科学大学講師, 95 年東京工業大学工学部助教授, 2000 年同大学理工学研究科准教授, 2016 年より同大学工学院システム制御系准教授. ロボティクス, 強化学習, 非線形制御の研究に従事 (工学博士).

佐 多 宏 太 (正会員)

2003 年名古屋大学大学院工学研究科機械情報システム工学専攻修士課程修了. 同年, トヨタ自動車 (株) に入社, 現在に至る. 自動車のパワーtrain 制御システムおよび, 開発環境の研究開発に従事.

松 永 彰 生 (正会員)

1993 年東京大学大学院工学系研究科機械工学専攻修士課程修了. 同年トヨタ自動車 (株) に入社, 現在に至る. パワーtrain 電子制御システムの研究・開発に従事.